



**Kauno technologijos universitetas**

Informatikos fakultetas

## Testavimo planas

T120B162 Programų sistemų testavimas

Pirmas laboratorinis darbas

---

**Autorius**

Gustas Klevinskas

**Akademinei grupei**

IFF-8/7

**Vadovas**

Doc. Šarūnas Packevičius

---

Kaunas, 2021

# Turinys

Turinys.....	2
1. Įvadas.....	3
1.1. Sistemos funkcijos.....	3
2. Testavimo apimtis.....	3
3. Testavimo prioritetai .....	4
4. Testavimo technikos .....	4
5. Testavimo aplinka .....	4
6. Testavimo tvarkaraštis.....	5
7. Testavimo rizikos .....	5
8. Testavimo scenarijai .....	5
8.1. Prisijungimas .....	5
8.2. Komentaro sukūrimas straipsnyje .....	6
8.3. Nuotraukos įkėlimas.....	6
8.4. Varžybų rezultatų paieška.....	7

# 1. Įvadas

Testavimui pasirinktas kuriamas individualus projektas moduliui „Saityno taikomųjų programų projektavimas“. Tai yra internetinis puslapis su REST API, skirtas Kauno sportinio šaudymo klubui dalintis įrašais apie renginius ir kitas veiklas (kurti straipsnius), nuotraukų iš renginių dalinimuisi, skelbti varžybų kalendorių ir jų rezultatus.

Puslapyje yra trys skirtingi naudotojų tipai:

- neregistruotas naudotojas;
- registruotas naudotojas;
- puslapio administratorius.

## 1.1. Sistemos funkcijos

Žemiau pateikti sistemos funkciniai reikalavimai išskirti kiekvienam naudotojų tipui.

Neregistruotas sistemos naudotojas galės:

1. Peržiūrėti straipsnius.
2. Peržiūrėti nuotraukas.
3. Peržiūrėti varžybų kalendorių ir rezultatus.
4. Susikurti paskyrą.

Registruotas sistemos naudotojas galės:

1. Peržiūrėti straipsnius.
2. Peržiūrėti nuotraukas.
3. Peržiūrėti varžybų kalendorių ir rezultatus.
4. Palikti komentarą po straipsniu.

Administratorius galės:

1. Peržiūrėti straipsnius.
2. Peržiūrėti varžybų kalendorių ir rezultatus.
3. Sukurti naujus straipsnius.
4. Įkelti nuotraukas.
5. Registruoti varžybas.
6. Įvesti konkrečių varžybų rezultatus.
7. Skirti ir atimti administratoriaus teises kitam registruotam naudotojui.

## 2. Testavimo apimtis

Bus testuojamos šios sistemos dalys:

- Web puslapis
- API

Naudojant naudotojo sąsają (web puslapį) turi būti patikrinti sistemos funkciniai reikalavimai, ar jie įgyvendinti. API pusėje turi būti patikrintas HTTP užklausų tikrinimas, ar atitinka web formų validaciją,

ir ar išlaikomas prisijungimo reikalavimas – pavyzdžiui, kad neregistruotas naudotojas negalėtų atlikti administratoriaus rolės veiksmų.

### 3. Testavimo prioritetai

Žemiau išvardintos skiltys nuo svarbiausios:

1. Funkciniai reikalavimai – ar įgyvendinti visi aukščiau paminėti reikalavimai;
2. Saugumas – ar prie funkcionalumų gali prieiti tik tą autorizacijos lygį turintys naudotojai;
3. Greitaveikos – ar naudotojo sąsaja veikia sklandžiai, ar serveris greitai atsako į HTTP užklausas.

### 4. Testavimo technikos

Sistemai testuoti pasitelktos šios priemonės:

1. Automatizuoti vienetų (*angl.* unit) testai – atsakingi už atskiros funkcijos veikimo patikrinimą.
2. Automatizuoti integraciniai testai – skirti patikrinti, kaip veikia atskiros funkcijos tarpusavyje.
3. Našumo testavimas – funkcionalumų, kaip nuotraukų įkėlimas ar peržiūra, varžybų kalendorius ir pan., greitaveika.
4. Saugos testai – skirti patikrinti, ar teisingai atsižvelgiama į naudotojų tipus.

### 5. Testavimo aplinka

Veikiantis web serveris bus patalpintas serveryje su šiais techniniais parametrais:

- Intel Core i7-4500U, 8 GB RAM, 256 GB SSD diskas.
- Arch Linux operacinė sistema.
- Nginx web serveris.
- ASP.NET Core 5
- Kuriamas tinklapis.

Darbinis kompiuteris:

- Intel Core i5-3320M, 8 GB RAM, 1 TB SSD diskas.
- Arch Linux operacinė sistema.
- Node 16 ir ASP.NET Core 5 SDK.

Naudotojo kompiuteris:

- Bet koks modernus kompiuteris ar telefonas.
- Prieiga prie interneto.
- Interneto naršyklė, palaikanti JavaScript.

## 6. Testavimo tvarkaraštis

Testavimo technika	Nuo	Iki
Vienetų testai	2021-09-15	2021-12-01
Integraciniai testai	2021-11-01	2021-12-01
Našumo testavimas	2021-11-01	2021-12-20
Saugos testai	2021-11-15	2021-12-20

## 7. Testavimo rizikos

Sėkmingam puslapio ištestavimui gresia šios rizikos:

- Lėtas sistemos realizavimas – programavimas gali užtrukti ilgiau nei planuota ir sistemos dalys, reikalingos testavimui pagal tvarkaraštį, gali būti dar nerealizuotos.
- Žmogiškieji ištekliai – pilnam programavimo ir testavimo darbų įvykdymui gali pritrūkti žmonių ar laiko.
- Sistemų trikdžiai – išorinės problemos, kaip interneto tiekimo trikdžiai serveryje.

## 8. Testavimo scenarijai

### 8.1. Prisijungimas

**Aprašas:** testuojamas prisijungimo forma.

**Prielaidos:** naudotojas turi susikūręs paskyrą, kurios el. paštas yra test@test.com, slaptažodis „test-pass“.

**Testavimo žingsniai:**

1. Atidaromas puslapis.
2. Paspaudžiamas mygtukas „Prisijungti“.
3. Pateiktoje formoje įvedamas el. paštas ir slaptažodis.

El. paštas	Slaptažodis	Tikimasis rezultatas
test@test.com	testpass	Naudotojas sėkmingai prijungiamas ir atidaromas prieš tai atidarytas svetainės puslapis.
test	<nėra>	Pateikiamas pranešimas prie el. pašto lauko „Įveskite el. pašto adresą“ ir prie slaptažodžio lauko – „Įveskite slaptažodį“.

test@test.com	wrongpass	Pateikiamas pranešimas prie „Prisijungti“ mygtuko: „Neteisingas el. pašto adresas ar slaptažodis“.
---------------	-----------	--

## 8.2. Komentaro sukūrimas straipsnyje

**Aprašas:** testuojamas komentaro sukūrimas straipsnyje.

**Prielaidos:** jau egzistuoja bent vienas straipsnis.

**Testavimo žingsniai:**

1. Atidaromas straipsnių sąrašas.
2. Atidaromas straipsnis.

Žemiau pateikta lentelė su skirtingų rolių testavimo variantais ir tikėti rezultatai.

Rolė	Veiksmai	Tikimasis rezultatas
Neprisijungęs naudotojas	<nėra>	Komentaro įvedimo laukas išjungtas ir po juo rodomas pranešimas „Komentarus gali palikti tik prisijungę naudotojai“.
Prisijungęs naudotojas (vardu „Tomas“)	Į komentaro laukelį įvedamas komentaras „Komentaras“.	Komentarų skiltis atsinaujina, naudotojas mato sukurtą komentarą su tekstu „Komentaras“, kaip autorius parašyta „Tomas“.
Administratorius (vardu „Petras“)	Į komentaro laukelį įvedamas komentaras „Komentaras“.	Komentarų skiltis atsinaujina, administratorius mato sukurtą komentarą su tekstu „Komentaras“, kaip autorius parašyta „Petras“. Prie vardo parašytas tekstas „(admin)“.

## 8.3. Nuotraukos įkėlimas

**Aprašas:** testuojamas nuotraukos įkėlimas prie straipsnio.

**Prielaidos:** naudotojas yra prisijungęs, yra užpildyta straipsnio kūrimo forma.

**Testavimo žingsniai:**

1. Paspaudžiamas mygtukas „Įkelti nuotrauką“.
2. Naudotojo įrenginyje atidaromas lankas, leidžiantis pasirinkti nuotrauką.

Failas	Failo dydis	Tikimasis rezultatas
Pasirenkamas nuotraukos failas.	4 MB	Nuotrauka įkeliama. Išsaugojus straipsnį nuotrauka rodoma straipsnio nuotraukų galerijoje.
Pasirenkamas tekstinis failas.	1 KB	Failas nepridedamas, parodomas pranešimas „Prašome įkelti nuotraukos failą“.

Pasirenkamas nuotraukos failas.	30 MB	Nuotraukos failas nepridedamas, parodomas pranešimas „Failo dydis turi būti mažesnis nei 8 MB“.
---------------------------------	-------	---

## 8.4. Varžybų rezultatų paieška

**Aprašas:** testuojamas varžybų dalyvio buvę rezultatai.

**Prielaidos:** asmuo, vardu pavarde „Vardenis Pavardenis“, dalyvavęs keliose varžybose.

**Testavimo žingsniai:**

1. Atidaromas varžybų puslapis.
2. Pasirenkama skiltis „Rezultatų paieška“.

Paieškos laukelis	Tikimasis rezultatas
<nieko neįvedama>	Prie paieškos laukelio parodomas pranešimas „Įveskite vardą pavardę“, paieška neatliekama.
Įvedama „Kitas Kitauskas“.	Paieškos rezultatų lentelėje rodomas pranešimas „Varžybų dalyvis nerastas“.
Įvedama „Vardenis Pavardenis“.	Paieškos rezultatų lentelėje rodomas „Vardenis Pavardenis“ ir jo praeitų varžybų pavadinimas, data, vieta bei rezultatai.

## 8.5. Komentaro sukūrimo saugumas

**Aprašas:** testuojama, ar gerai įgyvendintas rolių funkcionalumų atskyrimas.

**Prielaidos:** testuotojas turi paskyrą, prie kurios gali prisijungti, yra sukurtas straipsnis; HTTP užklausų siuntimo programą, pavyzdžiui, Postman.

**Testavimo žingsniai:**

1. Atidaromas Postman (ar atitinkama programa).
2. Padaroma užklausa su teisingais prisijungimo duomenimis į <serverio adresas>/API/login.
3. Išsisaugomas gautas JWT (JSON Web Token).
4. Padaroma POST užklausa į /API/posts/<straipsnio ID>/comments.

Header'is	Tikimasis rezultatas
Nėra Authorization header'io.	Grąžinamas 403 HTTP kodas.
Pateiktas teisingas, tačiau koreguotas header'is.	Grąžinamas 403 HTTP kodas.
Įvedamas iš API gautas tokenas.	Komentaras išsaugomas ir grąžinamas 201 HTTP kodas.

## 8.6. Aktyviausių komentatorių skiltis

**Aprašas:** administratoriaus turi prieigą prie statistikos, kuri rodo, top 10 aktyviausių komentatorių puslapyje. Atsižvelgiama ne tik į komentarų skaičių, bet ir kaip jie paskirstyti.

**Prielaidos:** testuotojas turi administratoriaus paskyrą ir yra prisijungęs.

**Testavimo žingsniai:**

1. Atidaromas statistikų puslapis.

Naudotojai ir komentarų skaičius	Tikimasis rezultatas
Nėra paliktas nė vienas komentaras.	Parodoma žinutė „Komentarų nerasta.“
Vardenis palikęs po 2 komentarus atskiruose straipsniuose, Tomas palikęs 10 komentarų viename straipsnyje.	Grąžinamas sąrašas: 1. Tomas – 10. 2. Vardenis – 4.
Vardenis palikęs po 2 komentarus atskiruose straipsniuose, Tomas palikęs 10 komentarų viename straipsnyje, Petras palikęs po 1 komentarą atskiruose 10 straipsnių.	Grąžinamas sąrašas: 1. Petras – 10. 2. Tomas – 10. 3. Vardenis – 4.